テーマ２・問３

今回は Bisection 問題に対応する特殊な形の連立方程式（とその確率モデル）について説明したが，メッセージ伝播法は，もっと自然な次のような連立方程式の生成モデルでも平均に有効に動くことが知られている．
どのような確率パラメータ p, q だとうまく動くかを実験し，その結果・解析・考察を述べよ．

今回はサンプルプログラムの **sample2mpass** と **sample2gwalk** を比較した．ひとまずは性能の比較，n を $\{100, 110, 120, 130, 140, 150, 160, 160, 170, 180, 190, 200\}$ に繰り返し設定し，p を $\{\frac{2}{n}, \frac{10}{n}\}$，$q$ を $\{0.0, 0.1, 0.3\}$ とおく，実験結果は下記である．

- Compiler:gcc version 4.2.1 (Based on Apple Inc. build 5658) (LLVM build 2336.9.00)
- CPU:1.7GHz Inter Core i5
- Memory:4GB 1333 MHz DDR3

表 1　sample2gwalk

(p, q)	$(\frac{2}{n}, 0.0)$	$(\frac{2}{n}, 0.1)$	$(\frac{2}{n}, 0.3)$	$(\frac{10}{n}, 0.0)$	$(\frac{10}{n}, 0.1)$	$(\frac{10}{n}, 0.3)$
time	0.02	0.02	0.02	0.02	0.02	0.02
t	5210	5977	6337	2187	4306	6555

表 2　sample2mpass

(p, q)	$(\frac{2}{n}, 0.0)$	$(\frac{2}{n}, 0.1)$	$(\frac{2}{n}, 0.3)$	$(\frac{10}{n}, 0.0)$	$(\frac{10}{n}, 0.1)$	$(\frac{10}{n}, 0.3)$
time	0.10	0.11	0.11	0.03	0.06	0.35
t	10925	10925	10925	161	684	10925

ここの t は各 p, q の組み合せの総反復数，time はコマンド **time** を用いて測った計算時間である．全体的に，greedy random walk のほうが反復が少なく，計算が早い．しかし，greedy random walk の計算時間と反復数は p, q と大きい差がないが，message passing のは p, q と強い相関性が見られる：q が小さければ小さいほど，反復が少ない，計算も早い．次は，n と反復数の関係を見つけるため，$p = \frac{10}{n}, q = 0.0$ に定め，n ごとの反復数を計算する．上記のように，message passing の t は，

表 3　n と反復数の関係

n	110	120	130	140	150	160	170	180	190	200
信念伝播法 t	15	15	15	15	15	16	17	16	17	23
Random walk t	215	166	379	463	633	678	607	641	858	903

n が増やしてもあんまり多くにならないが，random walk の t は n とともに増やしやすく，大規模の問題に適用できないと思う．

一方，メモリー消費を考えると，Random walk のソースは Message Passing のより一個の $O(n^2)$ 領域複雑度の配列が多いので，大きい n に対し，Message Passing のほうが効率がよいと見られる．